ABSTRACT

A system, method and apparatus is provided that splits a microprocessor load instruction into two (2) parts, a speculative load instruction and a check speculative load instruction. The speculative load instruction can be moved ahead in the instruction stream by the compiler as soon as the address and result registers are available. This is true even when the data to be loaded is not actually required. This speculative load instruction will not cause a fault in the memory if the access is invalid, i.e. the load misses and a token bit is set. The check speculative load instruction will cause the speculative load instruction to be retried in the event the token bit was set equal to one. In this manner, the latency associated with branching to an interrupt routine will be eliminated a significant amount of the time. It is very possible that the reasons for invalidating the speculative load operation are no longer present (e.g. page in memory is not present) and the load will be allowed to complete. Therefore, substantial gains in efficiency and resource utilization can be achieved by deferring the branch to recovery routines until after the speculative load is retried.

10045140 .031402